



Docket No. POU920020009US1

In re application of: Wayne A. SAWDON et al.

Serial No.: 10/077,371

Filed: February 15, 2002

For: WRITABLE FILE SYSTEM SNAPSHOT WITH DITTO ADDRESS FEATURE

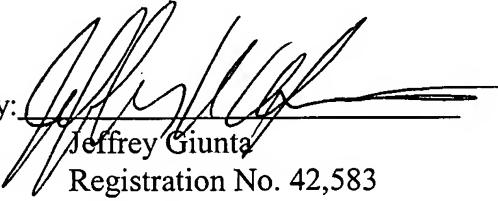
Mail Stop Appeal Brief- Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Transmitted herewith, in triplicate, is Appellants' Brief in support of its appeal to the Board of Patent Appeals and Interferences from the decision dated March 7, 2005, of the Examiner finally rejecting claims 1-30 of the above-referenced application.

- [] A petition for extension of time is enclosed.
- [X] The Commissioner is hereby authorized to charge payment in the amount of \$ 500.00 to cover the filing fee to Deposit Account No. 09-0463.
- [X] The Commissioner is hereby authorized to charge payment of any necessary fees associated with this communication, or credit any overpayment, to Deposit Account No. 09-0463.

Respectfully submitted,

By: 

Jeffrey Giunta

Registration No. 42,583

Date: September 7, 2005

FLEIT, KAIN, GIBBONS,
GUTMAN, BONGINI & BIANCO P.L.
One Boca Commerce Center
551 Northwest 77th Street, Suite 111
Boca Raton, Florida 33487
Telephone: (561) 989-9811
Facsimile: (561) 989-9812

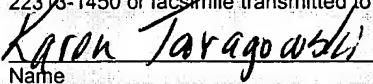
CERTIFICATE OF MAILING OR TRANSMISSION

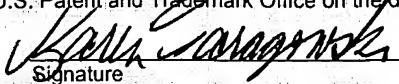
I hereby certify that this correspondence is being deposited with the United States Postal Service as First Class Mail in an envelope addressed to: MAIL STOP APPEAL BRIEF - PATENTS, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 or facsimile transmitted to the U.S. Patent and Trademark Office on the date shown below.

Name

Signature

Date





9/7/05



PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of:)
Wayne A. Sawdon, et al.)
Serial No.: 10/077,371)
Group Art Unit: 2164)
Filed: February 15, 2002)
Examiner: Belix M. ORTIZ)
For: WRITABLE FILE SYSTEM)
WITH DITTO ADDRESS FEATURE)

APPELLANTS' BRIEF UNDER 37 C.F.R. §1.192

MS – APPEAL BRIEF - PATENTS

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This Appellants' Brief is filed in response to a Final Office Action dated March 7, 2005 and a Notice of Appeal received July 7, 2005. Reconsideration of the Application, withdrawal of the rejections and allowance of the claims are respectfully requested.

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Mail Stop Appeal Brief-Patent, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

ON: 9/2/05 BY: John J. Magawski

09/13/2005 MAHMED1 00000049 090463 10077371

01 FC:1402 500.00 DA

I. REAL PARTY IN INTEREST

The real party in interest is International Business Machines (IBM) of Armonk, NY.

II. RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences.

III. STATUS OF CLAIMS

Claims 1-30 are pending. Claims 1-30 are rejected.

The Examiner's rejection of claims 1-30 is on appeal.

Attached hereto is an Appendix containing a copy of claims 1-30, which include the claims involved in this appeal.

IV. STATUS OF AMENDMENTS

No amendments were filed subsequent to the final rejection of March 7, 2005.

V. SUMMARY OF THE INVENTION

Preferred Embodiments of the present invention provide an improved apparatus, computer-readable medium and method for providing writable file system snapshots that require less processing than required by prior art systems. File system snapshots capture the state of a file system, i.e., preserve the state of data stored in the file system, at the particular point in time that the snapshot is captured. Specification, page 16, lines 25-28. Preferred embodiments of the present invention capture a snapshot by creating a sparse shadow inode file that is associated with the snapshot. The creation of the snapshot, including the creation of the shadow inode file, does not involve the allocation of any data blocks, but rather only establishes an inode for the shadow inode file that reflects that the shadow inode file has the same length as the inode file of the active file system. The disk addresses in this initially created shadow inode file are all set to the NULL, or zero, value to indicate that the data blocks for the shadow inode

have not yet been allocated. Specification, page 18, lines 8-26. The minimal processing described above to capture a snapshot advantageously requires only a brief interruption of the operation of the active file system – only long enough to establish the inode of the sparse shadow inode file and record its location in the superblock of the file system. Specification, page 19, lines 9-19.

After capture of a snapshot, active file system activity resumes. The active file system activity can include modifying attribute information, appending data to a data file or overwriting data in a data file. Modifying or deleting metadata or data within the active file system for the first time after the snapshot is captured results in the data in the file system at the time of snapshot capture being copied into the snapshot. FIG. 7A, Specification, page 20, line 12 through page 25, line 26. This results in data within a “snapshot” actually remaining within the active file system until the data in the active file system is modified and causes the snapshot to be initially “empty.” FIG. 8A, specification, page 26, lines 8-12. Retrieving data from a snapshot dataset involves examining the inode file of the snapshot dataset. Either no metadata or a “ditto value” is stored within an inode to indicate that the data has not been modified since the snapshot was captured and that the data is actually stored in the original file system. Specification, page 26, line 13 through page 27, line 23.

The preferred embodiments of the present invention further support multiple snapshots where the state of the active file system is captured at different times. Specification, page 30, lines 20-23. When a first snapshot has been captured and a subsequent snapshot is then captured, updates to the active file system are stored in the subsequent snapshot. Specification, page 31, lines 5-11. Retrieval of data from a particular snapshot then involves determining if the snapshot dataset indicates, through inferred references indicated by a lack of metadata or by containing “ditto addresses” for data blocks, that data is stored in a subsequent snapshot. FIGs. 10 and 11, specification, page 31, line 13 through page 34, line 18.

The preferred embodiments of the present invention also support multiple writable snapshots, where the data within a snapshot is able to be modified while

having the data that was contained within the snapshot moved into an earlier snapshot in order to preserve the data reflected by those earlier snapshots. Data to be operated on is either stored in the snapshot being modified or is retrieved from a subsequent snapshot. FIG. 7B, specification, page 35, line 20 through page 37, line 17.

VI. ISSUES

Whether claims 1-30 are anticipated or unpatentable by *Kazar et al* (U. S. Patent Publication No. 2002/0112022) in view of *Howard* (U.S. Patent Publication No. 2002/0078244) under 35 U.S.C. §102(e) or 35 U.S.C. §103(a).

VII. GROUPING OF CLAIMS

Group I: Claims 1-5, 9-13, and 17-21 stand or fall together.

Group II: Claims 6-8, 14-16, and 22-24, stand or fall together.

Group III: Claims 25-30, stand or fall together.

VIII. ARGUMENT

A. WHETHER CLAIMS 1-30 ARE ANTICIPATED OR UNPATENTABLE OVER KAZAR IN VIEW OF HOWARD

In the Examiner's Final Office Action of March 7, 2005, the Examiner rejected claims 1-30 under 35 U.S.C. § 102(e) as being anticipated *Kazar et al* (U. S. Patent Publication No. 2002/0112022) (Hereinafter *Kazar*) in view of *Howard* (U.S. Patent Publication No. 2002/0078244) (Hereinafter *Howard*). Although the Examiner's Final Office Action states that the rejection is under 35 U.S.C. §102(e), a proper rejection requires that a single reference teach (i.e., identically describe) each and every element of the rejected claims.¹ The

¹ See MPEP §2131 (Emphasis Added) "A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987). "The identical invention must be shown in as complete detail as is contained in the ...

Appellants assert that the combination of two references, i.e., the *Kazar* and *Howard* references, requires the rejection to come under 35 U.S.C. §103(a). A proper rejection under 35 U.S.C. §103 expressly requires that obviousness or non-obviousness be determined for the claimed subject matter “as a whole,” and the key to proper determination of the differences between the prior art and the present invention is giving full recognition to the invention “as a whole.” The following remarks discuss differences between the cited prior art and the claimed invention and indicates that rejection of the pending claims is improper under either 35 U.S.C. § 102(e) or 35 U.S.C. § 103.

The Appellants respectfully assert that the *Kazar* and *Howard* references, taken either alone or in combination with one another, do not teach or suggest the claimed limitations of: “determining if an inode to be modified in the specified snapshot is an empty inode; copying, in response to determining a inode to be modified is an empty inode, metadata corresponding to the inode to be modified into the inode to be modified; writing the metadata into a next oldest file system snapshot; and modifying the metadata copied into the inode to be modified” as is set forth for the presently claimed invention. These cited reference further do not teach or suggest “accessing a specified file system snapshot in a plurality of file system snapshots, wherein the specified file system snapshot comprises at least one inode comprising at least one ditto address, wherein the at least one ditto address refers to a data block that has a disk address in an inode associated with one of a more recent snapshot and a source file system” or “determining if a data block to be modified is referenced by a ditto address in an inode of the specified file system snapshot” as is also set forth for the presently claimed invention.

Group I: Claims 1-3, 5, 9-11, 13, 17-19, and 21

The Appellants respectfully suggest using claim 1 as representative of the Group I claims. To begin, the Appellants traverse the Examiner’s assertion in the last paragraph of page 2 and the first two paragraphs of page 3 of the Office

claim.”

Action dated March 7, 2005, that the Kazar reference teaches the following limitations of independent claim 1:

- 1) determining if an inode to be modified in the specified snapshot is an empty inode;
- 2) copying, in response to determining a inode to be modified is an empty inode, metadata corresponding to the inode to be modified into the inode to be modified;
- 3) writing the metadata into a next oldest file system snapshot; and
- 4) modifying the metadata copied into the inode to be modified.

With regards to the limitations of “determining if an inode to be modified in the specified snapshot is an empty inode” and “copying, in response to determining the inode to be modified is an empty inode, metadata corresponding to the inode to be modified into the inode to be modified,” the Appellants assert that in considering this claim “as a whole,” the Kazar reference makes no mention of, and therefore does not teach or suggest, this combination of “determining” and “copying, in response to determining.” With regards to the determining limitation, the Examiner cites paragraphs 82 and 84 of Kazar. Office Action Dated March 7, 2005, page 2, penultimate paragraph. These paragraphs discuss creating read-only replication of a volume and identifying differences between an existing replica and a new replica in order to propagate differences to a replica. Kazar, paragraphs 82 and 84. The Appellants assert that the Kazar reference does not teach or suggest “determining if an inode to be modified in the specified snapshot is an empty inode.” As discussed below, the Appellants respectfully assert that the Kazar reference never discusses or includes a teaching of “empty inodes,” especially in the context of the method set forth in the Group I claims when considering that claim “as a whole.” For example, an earlier limitation specifies that “an empty inode indicates that metadata corresponding to the empty inode is contained in one of a more recent snapshot and a source file system.” Further, the Appellants are unable to find any part of the Kazar reference that teaches, discusses, or suggests making any determinations regarding inodes in any snapshots. Therefore, the Kazar reference is not able to

disclose “determining if an inode to be modified in the specified snapshot is an empty inode” as is set forth in the Group I claims. As this determining is not taught or suggested, the Kazar reference cannot teach “copying, in response to determining the inode to be modified is an empty inode.”

With regards to “copying, in response to determining the inode to be modified is an empty inode...,” the Examiner cites Figures 1 and 8-9 of Kazar, as well as paragraphs 20-21, 73 and 85. The Appellants are unable to find any discussion of “empty inodes” in these cited portions of the Kazar reference, or in any portion of the Kazar reference. As discussed above, the Group I claims state that “the inode to be modified” is “in the specified snapshot.” The Appellants are unable to find any teaching in the Kazar reference that any “copying” is performed “in response to” any characteristic of an inode in a snapshot. The Kazar reference is completely silent on “copying” and simply does not suggest or teach “copying, in response to determining an inode to be modified is an empty inode” as is set forth in the Group I claims.

The Appellants are further unable to identify where the Kazar reference teaches “writing the metadata into a next oldest file system snapshot.” The Examiner cites Kazar, paragraph 69. Office Action dated March 7, 2005, page 3, first paragraph. This cited portion of Kazar describes clones of a data storage volume and states that “the clone itself is a read-only volume and can not be modified.” Kazar, paragraph 69. This is clearly a teaching away of the claimed invention, which is directed to modifying the snapshot and specifies “writing” to the snapshot. Further, the cited portion of Kazar describes a clone of an active file system. The Appellants respectfully assert that the Kazar reference does not teach or suggest “writing the metadata into a next oldest file system snapshot” as is set forth by the Group I claims, since the snapshots of the Kazar system are not modified, thereby requiring “writing the metadata into a next oldest file system snapshot.”

The Appellants assert that the “read-only” snapshots of Kazar are not compatible with “writing the metadata into a next oldest file system snapshot” and that modification of Kazar to operate with the presently claimed invention would

lead to an inoperable system. If references taken in combination would produce a “seemingly inoperative device,” such references have been held to teach away from the combination and thus cannot serve as predicates for a *prima facie* case of obviousness. *In re Sponnoble*, 405 F.2d 578, 587, 160 USPQ 237, 244 (CCPA 1969) (references teach away from combination if combination produces seemingly inoperative device); see also *In re Gordon*, 733 F.2d 900, 902, 221 USPQ 1125, 1127 (Fed. Cir. 1984) (inoperable modification teaches away).

The Appellants further assert that the Kazar teaching of “read-only” snapshots precludes a teaching of “modifying the metadata copied into the inode to be modified” as is asserted by the Examiner. Office Action dated March 7, 2005, page 3, second paragraph, citing Kazar, paragraph 55. The Appellants point out that the “inode to be modified” is specified in another limitation of this claim as being “in the specified snapshot. The Appellants point out that the cited portion of Kazar discusses renaming a file within the active file system, not within a snapshot. The Appellants point out that the cited portion of Kazar does not teach or suggest modifying a snapshot. The Appellants further assert that the above cited portion of Kazar teaches away from modifying data, including inode data, within a snapshot by stating that “the clone itself is a read-only volume and can not be modified.” Kazar, paragraph 69. Where the prior art *points away* from the combination, modification or substitution of which is the premise of the PTO’s alleged *prima facie* case of obviousness, there likewise is a built-in traversal of the rejection. See *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988).²

² The Federal Circuit held a reference did not render the claimed combination *prima facie* obvious because *inter alia*, the Examiner ignored material, claimed temperature limitations which were absent from the reference. See MPEP §2143.01 In *In re Fine*, the claims were directed to a system for detecting and measuring minute quantities on nitrogen compounds comprising a gas chromatograph, a converter which converts nitrogen compounds into nitric oxide by combustion, and a nitric oxide detector. The primary reference disclosed a system for monitoring sulfur compounds comprising a chromatograph, combustion means, and a detector, and the secondary reference taught nitric oxide detectors. The examiner and Board asserted that it would have been within the skill of the art to substitute one type of detector for another in the system of the primary reference, however the court found there was no support or

With regards to the Howard reference, the Appellants respectfully traverse the Examiners assertions as to the teachings of the Howard reference. The Examiner states that the Howard reference teaches "accessing a specified file system snapshot in a plurality of file system snapshots, wherein the specified file system snapshot comprises at least one empty inode, wherein an empty inode indicates that metadata corresponding to the empty inode is contained in one of a more recent snapshot and a source file system." Office Action dated March 7, 2005, page 3, fourth paragraph. The Examiner cites Howard, paragraphs 7 and 24.

The Appellants respectfully assert that the Howard reference does not teach "accessing a specified file system snapshot in a plurality of file system snapshots" as is asserted by the Examiner. The Appellants fail to see where Howard teaches or suggests "a plurality of file system snapshots" as is asserted by the Examiner. Howard reference discusses a "copy on write" protocol, but that protocol is only described in the context of updating a file in an active file system by performing modifications in temporary storage and atomically updating the active file system. Howard, Abstract and paragraph 20. The Appellants respectfully assert that the Howard reference does not contemplate any "snapshots" of the type specified in the context of the Group I claims and that the Howard reference cannot teach "accessing a specified file system snapshot in a plurality of file system snapshots."

The Appellants further respectfully assert that the Howard reference does not teach or suggest "wherein an empty inode indicates that metadata corresponding to the empty inode is contained in one of a more recent snapshot and a source file system." Office Action dated March 7, 2005, page 3, 4th paragraph. The Examiner apparently cites paragraph 24 of Howard as a teaching that objects can contain zero bytes, and that files are objects, therefore files can contain zero bytes, and therefore a file can be empty. The Appellants respectfully assert a teaching of an empty file is not necessarily a teaching of an empty inode for a file. The Appellants assert that in a conventional file system,

explanation of this conclusion and reversed.

an empty inode cannot represent a file since there would be no file name or other data to even identify or refer to the file. The Appellants respectfully assert that a file in a conventional file system, such as is taught by Kazar and Howard, with an empty inode would not have any metadata associated with it. The Appellants assert that a “file” in a conventional file system with an empty inode would be non-existent and that such a non-existent file is not consistent within the context of the Group I claims, when that claim is considered “as a whole.” For example, this limitation explicitly recites that “an empty inode indicates that metadata corresponding to the empty inode is contained in one of a more recent snapshot and a source file system.” This context precludes an empty inode in a conventional file system since metadata corresponding to the empty inode is contained elsewhere, and therefore exists.

The Appellants further assert that the Howard reference does not explicitly teach an empty inode as having any useful purpose. The Appellants further assert that the Howard reference does not teach or suggest, either alone or in any combination with the Kazar reference or any other cited prior art reference, that an empty inode is used to indicate anything. The Appellants assert that any teaching or suggestion by Howard of an empty inode is at best inferential and that no specific use or purpose of any kind is taught or suggested by the Howard reference for such an empty inode. The Appellants therefore assert that the Howard reference cannot teach or suggest that “an empty inode indicates that metadata corresponding to the empty inode is contained in one of a more recent snapshot and a source file system” as is claimed by the Group I claims.

With further regards to the Group I claims, the Appellants further traverse an Examiner’s assertion, made with regards to independent claim 9 of the Group I claims, that Kazar “teaches a system for modifying a file system snapshot.” Office Action dated March 7, 2005, page 6, penultimate paragraph, citing Kazar, paragraph 100. The cited portion of Kazar teaches capturing a snapshot, and then using a read-only volume to contain that snapshot and thereby allow multiple users to access data within that snapshot. The Appellants respectfully

assert that this is not a teaching of "modifying a file system snapshot" as is recited by the Group I claims.

Group II: Claims 6-8, 14-16, and 22-24

The Appellants respectfully suggest selecting independent claim 6 as representative of the Group II claims. To begin, the Appellants traverse the Examiner's assertion that Kazar teaches "accessing a specified file system snapshot in a plurality of file system snapshots, wherein the specified file system snapshot comprises at least one inode comprising at least one ditto address, wherein the at least one ditto address refers to a data block that has a disk address in an inode associated with one of a more recent snapshot and a source file system." Office Action dated March 7, 2005, page 5, penultimate paragraph, citing Kazar paragraphs 68-69 and 71. The Appellants are unable to identify where Kazar teaches "wherein the at least one ditto address refers to a data block that has a disk address in an inode associated with one of a more recent snapshot and a source file system." The Appellants respectfully assert that the Kazar reference is completely silent as to any type of "ditto address" as is specified in the Group II claims and further defined in the Appellants' specification at, for example, page 28, lines 16-24 and page 34, lines 4-26.

The Appellants further respectfully traverse the Examiner's assertion that the Kazar reference teaches "determining if a data block to be modified is referenced by a ditto address in an inode of the specified file system snapshot." Office Action dated March 7, 2005, page 5, last paragraph, citing Kazar, paragraph 68. The Cited portion of Kazar describes indirect blocks in a file system, but fails to teach any time of "determining" and in particular fails to teach "determining if a data block to be modified is referenced by a ditto address in an inode of the specified file system snapshot." Since this determining is not taught, the Appellants respectfully assert that Kazar cannot, and does not, teach doing anything in response to such determining, such as the recited claim limitation of "copying, in response to determining the data block to be modified is referenced

by a ditto address in an inode of the specified file system snapshot, the data block to be modified into the specified snapshot."

The Appellants further assert that the other limitations of claim 6 regarding 1) copying the data block; and 2) modifying the data block, are not taught or suggested by the Kazar reference for reasons similar to those discussed above with regards to the similar limitations of the Group I claims.

Group III: Claims 25-30, stand or fall together.

The Appellants respectfully suggest use of independent claim 25 as representative of the Group IV claims. The Appellants point out that the Group III claims include a combination of the limitations of the Group I claims and the Group II claims. The Appellants therefore assert that the Group III claims distinguish over the Kazar and Howard references, taken either alone or in any combination with one another, for at least the same reasons discussed above for either or both of the Group I claims and/or the Group II claims.

IX. CONCLUSION

For the reasons stated above, Appellants respectfully contend that each claim is patentable. Therefore, reversal of all rejections is courteously solicited.

Respectfully submitted,

By: 
Jeffrey N. Giunta (Reg. No. 42,583)
Attorney for Appellants

Fleit, Kain, Gibbons, Gutman & Bongini
One Boca Commerce Center, Suite 111
551 N.W. 77th Street
Boca Raton, FL 33487
Tel. (561) 989-9811
Fax (561) 989-9812

IX. APPENDIX

1. A method for modifying a file system snapshot, comprising:
 - accessing a specified file system snapshot in a plurality of file system snapshots, wherein the specified file system snapshot comprises at least one empty inode, wherein an empty inode indicates that metadata corresponding to the empty inode is contained in one of a more recent snapshot and a source file system;
 - determining if an inode to be modified in the specified snapshot is an empty inode;
 - copying, in response to determining the inode to be modified is an empty inode, metadata corresponding to the inode to be modified into the inode to be modified;
 - writing the metadata into a next oldest file system snapshot; and
 - modifying the metadata copied into the inode to be modified.
2. The method of claim 1, wherein the metadata includes any one of:
 - at least one shadow inode, at least one indirect block referenced by a shadow inode and at least one data block referenced by a disk address in an indirect block.
3. The method of claim 1, further comprising:
 - updating the metadata of the specified file system snapshot in accordance with modifications to at least one source file corresponding to the specified file system snapshot.
4. The method of claim 1, further comprising:
 - accessing a next most recent file system snapshot;
 - copying a next metadata of the next most recent file system snapshot, wherein the next metadata includes any one of:

at least one shadow inode and at least one data block referenced by a disk address in a shadow inode; and
at least one shadow inode, and
writing the next metadata which have been copied to the specified file system snapshot.

5. The method of claim 4, further comprising writing the next metadata into the next oldest file system snapshot.

6. A method for modifying snapshot data, comprising:
accessing a specified file system snapshot in a plurality of file system snapshots, wherein the specified file system snapshot comprises at least one inode comprising at least one ditto address, wherein the at least one ditto address refers to a data block that has a disk address in an inode associated with one of a more recent snapshot and a source file system;
determining if a data block to be modified is referenced by a ditto address in an inode of the specified file system snapshot;
copying, in response to determining the data block to be modified is referenced by a ditto address in an inode of the specified file system snapshot, the data block to be modified into the specified snapshot;
copying the data block to be modified into a next oldest file system snapshot; and
modifying the data block copied into the specified snapshot.

7. The method of claim 6, further comprising:
wherein the copying a data block to be modified comprises reading an indirect block referenced by the disk address and at least one data block referenced by at least one disk address in the indirect block.

8. The method of claim 6, further comprising:
 - wherein the copying a data block to be modified comprises accessing one of a more recent snapshot dataset and the source file system to read the metadata.
9. A system for modifying a file system snapshot, comprising:
 - means for accessing a specified file system snapshot in a plurality of file system snapshots, wherein the specified file system snapshot comprises at least one empty inode, wherein an empty inode indicates that metadata corresponding to the empty inode is contained in one of a more recent snapshot and a source file system;
 - means for determining if an inode to be modified in the specified snapshot is an empty inode;
 - means for copying, in response to determining the inode to be modified is an empty inode, metadata corresponding to the inode to be modified into the inode to be modified;
 - means for writing the metadata into a next oldest file system snapshot; and
 - means for modifying the metadata copied into the inode to be modified.
10. The system of claim 9, wherein the metadata includes any one of:
 - at least one shadow inode and at least one data block referenced by a disk address in a shadow inode.
11. The system of claim 9, further comprising:
 - means for updating the metadata of the specified file system snapshot in accordance with modifications to at least one source file corresponding to the specified file system snapshot.

12. The system of claim 9, further comprising:
 - means for accessing a next most recent file system snapshot;
 - means for copying a next metadata of the next most recent file system snapshot, wherein the next metadata includes any one of:
 - at least one shadow inode and at least one data block referenced by a disk address in a shadow inode; and
 - at least one shadow inode, and
 - means for writing the next metadata which have been copied to the first specified file system snapshot.
13. The system of claim 12, further comprising writing the next metadata into the next oldest file system snapshot.
14. A system for retrieving snapshot data, comprising:
 - means for accessing a specified file system snapshot in a plurality of file system snapshots, wherein the specified file system snapshot comprises at least one inode comprising at least one ditto address, wherein the at least one ditto address refers to a data block that has a disk address in an inode associated with one of a more recent snapshot and a source file system;
 - means for determining if a data block to be modified is referenced by a ditto address in an inode of the specified file system snapshot;
 - means for copying, in response to determining the data block to be modified is referenced by a ditto address in an inode of the specified file system snapshot, the data block to be modified into the specified snapshot;
 - means for copying the data block to be modified into a next oldest file system snapshot; and
 - means for modifying the data block copied into the specified snapshot.

15. The system of claim 14, further comprising:
means for reading an indirect block referenced by the disk address and at least one data block referenced by at least one disk address in the indirect block.
16. The system of claim 14, further comprising:
means for accessing one of a more recent snapshot dataset and the source file system to read the metadata.
17. A computer readable medium including computer instructions for updating a file system snapshot, the computer instructions comprising instructions for:
accessing a specified file system snapshot in a plurality of file system snapshots, wherein the specified file system snapshot comprises at least one empty inode, wherein an empty inode indicates that metadata corresponding to the empty inode is contained in one of a more recent snapshot and a source file system;
determining if an inode to be modified in the specified snapshot is an empty inode;
copying, in response to determining the inode to be modified is an empty inode, metadata corresponding to the inode to be modified into the inode to be modified;
writing the metadata into a next oldest file system snapshot; and
modifying the metadata copied into the inode to be modified.
18. The computer readable medium of claim 17, wherein the metadata includes any one of:
at least one shadow inode and at least one data block referenced by a disk address in a shadow inode.

19. The computer readable medium of claim 17, further comprising instructions for:

 updating the metadata of the specified file system snapshot in accordance with modifications to at least one source file corresponding to the specified file system snapshot.

20. The computer readable medium of claim 17, further comprising instructions for:

 accessing a next most recent file system snapshot;

 copying a next metadata of the next most recent file system snapshot, wherein the next metadata includes any one of:

 at least one shadow inode and at least one data block referenced by a disk address in a shadow inode; and

 at least one shadow inode, and

 writing the data contents which have been copied to the specified file system snapshot.

21. The computer readable medium of claim 20, further comprising writing the next metadata into the next oldest file system snapshot.

22. A computer readable medium including computer instructions for retrieving snapshot data, the computer instructions comprising instructions for:

accessing a specified file system snapshot in a plurality of file system snapshots, wherein the specified file system snapshot comprises at least one inode comprising at least one ditto address, wherein the at least one ditto address refers to a data block that has a disk address in an inode associated with one of a more recent snapshot and a source file system;

determining if a data block to be modified is referenced by a ditto address in an inode of the specified file system snapshot;

copying, in response to determining the data block to be modified is referenced by a ditto address in an inode of the specified file system snapshot, the data block to be modified into the specified snapshot;

copying the data block to be modified into a next oldest file system snapshot; and

modifying the data block copied into the specified snapshot.

23. The computer readable medium of claim 22, further comprising instructions for:

wherein the copying a data block to be modified comprises reading an indirect block referenced by the disk address and at least one data block referenced by at least one disk address in the indirect block.

24. The computer readable medium of claim 22, further comprising instructions for:

wherein copying a data block to be modified comprises accessing one of a more recent snapshot dataset and the source file system to read the metadata.

25. A system for updating a file system snapshot, comprising:
a first file system snapshot in a plurality of file system snapshots, wherein
the first file system snapshot includes data contents;
data contents of the first file system snapshot, wherein the data contents
comprises at least one of an empty inode and an inode comprising at least one
ditto address,

wherein an empty inode indicates that metadata corresponding to
the empty inode is contained in one of a more recent snapshot, and a
source file system, and

wherein a ditto address refers to a data block that has a disk
address in an inode associated with one of the more recent snapshot, and
a source file system; and
means for writing the data contents to a next oldest file system snapshot.

26. The system of claim 25, wherein the data contents includes any one of:
at least one shadow inode, at least one indirect block referenced by a
shadow inode and at least one data block referenced by a disk address in an
indirect block; and
at least one shadow inode.

27. The system of claim 25, further comprising:
a next most recent file system snapshot;
data contents of the next most recent file system snapshot, wherein the
data contents includes any one of:
at least one shadow inode and at least one data block referenced
by a disk address in a shadow inode; and
at least one shadow inode, and
means for writing the data contents to the first file system snapshot.

28. A system for retrieving snapshot data, comprising:
 - a shadow inode in a first snapshot dataset corresponding to a source file, the shadow inode comprising at least one of an empty inode and an inode comprising at least one implied reference,
 - wherein an empty inode indicates that an metadata corresponding to the empty inode is contained in one of a more recent snapshot, and a source file system, and
 - wherein an implied reference refers to a data block that has a disk address in an inode associated with one of the more recent snapshot, and a source file system; and
 - a next most recent snapshot dataset.
29. The system of claim 28, further comprising:
 - an indirect block referenced by the disk address; and
 - at least one data block referenced by at least one disk address in the indirect block.
30. The system of claim 28, further comprising:
 - a next most recent snapshot dataset having a different ancestor as the first snapshot dataset.



09/12/05

zta
AF

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application No.: **09/604,824**

Group Art Unit: **2614**

Filing Date: **June 27, 2000**

Examiner: **John Manning**

Applicant: **Jeffrey C. Shroeder**

Title: **System and Method for Integrating Weather Data into Television Broadcast**

ATTORNEY DOCKET: **FL001**

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

CERTIFICATE OF EXPRESS MAIL

I hereby certify that this Amended Appellant's Brief and Response to Notification of Non-Compliance of Appeal Brief with 37 C.F.R. §1.192(c) in the above-identified application, along with any paper referred to as being attached or enclosed, is being deposited with United States Postal Service with sufficient postage as Express Mail on September 9, 2005 Express Mail Label No. ET377744944US.

We believe that no payment is due with this submission but in the event that this is not the case, the Commissioner for Patents is hereby authorized to charge such payment to Deposit Account No. 19-3790 for any fees associated with this communication.

Joseph P. C. Cifelli
(Print Name)

Joseph P. C. Cifelli
(Signature)